# An Evolutionary Approach for Balancing Effectiveness and Representation Level in Gene Selection

*Nicoletta Dessì, Department of Mathematics and Computer Science, Università degli Studi di Cagliari, Cagliari, Italy*

*Barbara Pes, Department of Mathematics and Computer Science, Università degli Studi di Cagliari, Cagliari, Italy*

*Laura Maria Cannas, Department of Mathematics and Computer Science, Università degli Studi di Cagliari, Cagliari, Italy*

## ABSTRACT

*As data mining develops and expands to new application areas, feature selection also reveals various aspects to be considered. This paper underlines two aspects that seem to categorize the large body of available feature selection algorithms: the effectiveness and the representation level. The effectiveness deals with selecting the minimum set of variables that maximize the accuracy of a classifier and the representation level concerns discovering how relevant the variables are for the domain of interest. For balancing the above aspects, the paper proposes an evolutionary framework for feature selection that expresses a hybrid method, organized in layers, each of them exploits a specific model of search strategy. Extensive experiments on gene selection from DNA-microarray datasets are presented and discussed. Results indicate that the framework compares well with different hybrid methods proposed in literature as it has the capability of finding well suited subsets of informative features while improving classification accuracy.*

*Keywords:* *Bioinformatics, Feature Selection, Genetic Algorithms, High-Dimensional Data, Microarray Data Analysis*

## INTRODUCTION

Feature selection is one of the important and frequently used techniques in data mining (Chandrashekar & Sahin, 2014). It reduces the number of features, removes irrelevant, redundant, or noisy data, and improves mining performance such as predictive accuracy and result comprehensibility.

The goodness of selected features is usually measured by an evaluation criterion that strongly affects results, i.e. an optimal set of

DOI: 10.4018/jitr.2015040102

features selected using one criterion may not be optimal according to another criterion. Despite the work on developing criteria for evaluating the quality of results in feature selection algorithms (Kumar & Minz, 2014), the choice of the algorithm appropriate for classification problems remains difficult. It has been argued (Liu & Yu, 2005) that the more feature selection algorithms available, the more difficult it is to select a suitable one for a classification task.

While there is no agreement about the definition of the mathematical statement of the problem (Guyon & Elisseeff, 2003), two major factors seem to be particularly important in designing a suitable algorithm for feature selection in a classification task: improving the predictive accuracy and providing better understanding of the underlying concept that generated the data. We denote the above factors as the *effectiveness* and the *representation level* of the feature selection process.

Specifically, the effectiveness deals with selecting the minimum set of variables that maximize the accuracy of a classifier and the representation level concerns discovering how relevant the variables are for the considered domain.

In more detail, the effectiveness attempts to capture the performance aspect of classification. From this point of view, the major challenge is finding a minimum subset of features that are useful to the prediction. Thus, this aspect is central for classification problems in which accuracy is of primary concern: the more effective the feature selection, the better the performance of the resulting classifier.

The representation level reflects the explanatory power of the selected features in representing essential knowledge about the application domain. The focus is on discovering all the variables suited to the reality that we are trying to represent, deciding how relevant and informative they are. Under this paradigm, the feature selection process privileges the usefulness of the features in representing the application domain i.e. the degree of exactness with which the representation fits the reality.

Research efforts have produced methods that place the emphasis at different times on the effectiveness or on the representation level (Tang, Alelyani, & Liu, 2014). Among the broadly used methods, *rankers* evaluate the discriminative power of features with regard to the class labels of samples by looking only at the intrinsic properties of the data. Thus, rankers emphasize the representation level giving as output a list where features are ordered based on their relevance for the classification task at hand.

Leveraging on rankers, *filter* methods strive to improve the effectiveness by selecting a certain number of highest ranked features for the purpose of classification. However, because the number used is somewhat arbitrary, features selected under this approach depend on an "a priori" choice with little support for determining how many features should be chosen for classification. Moreover, filters do not take into account the classifier to be applied.

In contrast to the filter approach, *wrapper* methods adopt a paradigm in which the main emphasis is on selecting features during the process of classification. Different subsets of features are generated by using a search algorithm and then are evaluated by training and testing a specific classification model. As the whole process aims to optimize the accuracy of the particular classifier, the central aspect in selecting features is the effectiveness rather than improving the representation level.

Recently, *hybrid* approaches have attempted to take advantage of the above methods by exploiting their different evaluation criteria in different search stages (Saeys, Inza, & Larranaga, 2007). However, to the best of our knowledge, it remains a neglected issue the formulation of feature selection methods that place the emphasis on balancing effectiveness and representation level.

This paper gives a contribution in this direction by proposing a framework for intelligent feature selection that aims not only to achieve good classification accuracy but also to discover different subsets of features relevant

for the application domain and able to represent it properly. Based on a genetic algorithm, the framework expresses a hybrid method organized in layers each of ones determines a model of search strategy by privileging, in turn, the effectiveness or the representation level.

The proposed framework takes a paradigm shift from other hybrid models that try to accommodate the use of filter and wrapper approaches according to the specific application domain. Guided by the motivation of discovering a potentially high number of predictors, the framework defines different subspaces of features for searching whereas existing hybrid models usually provide a single subspace.

To evaluate the proposed framework we conducted experiments on high dimensional biological data. Specifically, we considered DNA-microarray datasets which store genetic profiles of cancerous and normal tissues: here each feature expresses the level of expression of a specific gene, and the number of genes is in the order of thousands. The high dimensionality, coupled with the small sample size (typically a few dozens of samples), requires appropriate data mining strategies for selecting groups of genes (predictors) that are useful for understanding the cancer and do help determine more accurate diagnosis, prognosis as well as treatment planning (Bolón-Canedo, Sánchez-Maroño, Alonso-Betanzos, Benítez, & Herrera, 2014).

Gene selection from micro-array data is a significant example of application domain where the effectiveness of selected features should be evaluated in conjunction with their representation level. Although the abundance of features in gene expression data, it has been anticipated that only a limited number of them are informative for prognostic purpose about cancer. Moreover, only a small number of genes taken from a large expression dataset can be tested for clinical relevance. Thus, the emphasis is on identifying the smallest subset of genes that are potentially relevant for cancer prediction i.e. special attention is given to the effectiveness. However, besides the accuracy of the classification process, a biologist could discard the best predictor because it has a high biological cost to be validated or it reveals some obvious or previous knowledge. Hence, biologists may be interested in identifying different subsets of marker genes that contribute in a complementary way to best explain a given pathology. This goal makes relevant to achieve a good representation level.

In this work, we validated our framework on four DNA-microarray datasets. Experimental results compare well with different hybrid methods proposed in literature and show that our approach is robust and effective in finding small subsets of informative features with high classification accuracy and suitable representation level.

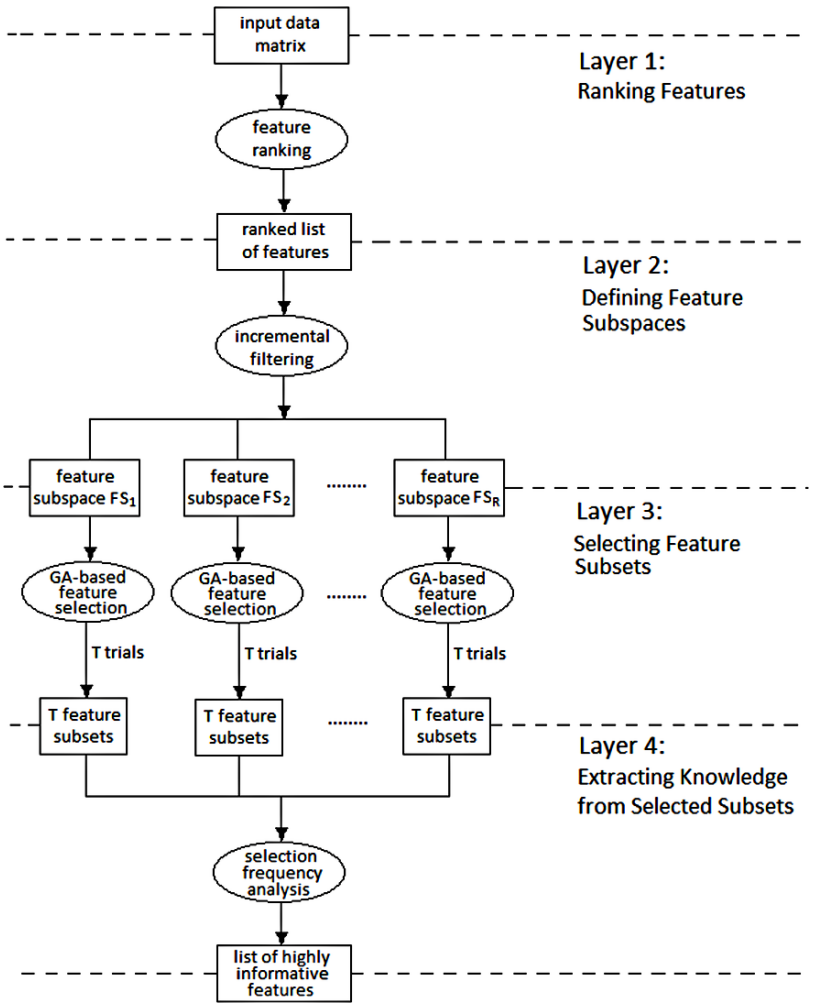
## THE PROPOSED FRAMEWORK

In this section, we present the framework we propose. As Figure 1 shows, the framework consists of four layers. Layers are ordered by increasing complexity of the learning process, from the dataset initially provided to the features ultimately selected. Each layer organizes a class of methods to transform data provided by the previous layer into some new form of information for future use by the next layer.

Specifically, the first layer considers ranking features to provide a general representation level of the application domain. At the second level, a filter provides different features subsets, namely feature subspaces. At the next layer, these subspaces are explored by a wrapper that uses a genetic algorithm (GA) as a search strategy. The availability of different feature subspaces is intended to enhance the search capability of the genetic algorithm in discovering sets of potential useful features in each subspace. Finally, at the last layer, potential useful features are evaluated to extract knowledge about the application domain.

As each layer refers to a general class of methods, the framework is independent on the choice of the algorithms for its implementation

*Figure 1. The framework*




i.e., at each layer, the tasks are not associated to a single method, but rather to a general class of methods.

## The Four Layers

In more detail, starting from the input data matrix, the first task is intended to scoring individual features according to their discriminative power, i.e. their capacity of separating the classes. It consists of ranking features and results in an ordered list where features appear in descending order of relevance.

At the layer 2, the next task is crucial as it defines subspaces of features where the GA will search the optimal feature subsets i.e. the best predictors. This is performed by a filter approach. Starting from the first P features of the ordered list that was produced at the previous layer, nested subsets of features of increasing size are constructed by progressively adding features (less and less correlated with the target).

It results in a sequence of R feature subspaces:

$$FS_1 \subset FS_2 \subset \ldots \subset FS_R$$

where the first space ($FS_1$) includes the first P top-ranked features, the second space ($FS_2$) includes the first 2·P features, etc. Denoting with N the dimension of the feature subspace, one obtains: N = i·P, i = 1, 2, …, R.

Although containing a subset of potentially informative features, each single subspace $FS_i$, i = 1, 2, …, R, cannot be considered a good predictor because its features may be mutually correlated. As such, additional work is needed for refining the above subsets by removing redundant features in order to devise more accurate and small-sized predictors. This is done at the third layer of the framework where a wrapper applies a GA for exploring and discovering optimal predictors in each subspace.

In more detail, a population of individuals (i.e. different features subsets) is randomly initialized from each single subspace. The individuals are first evaluated by a fitness function that is designed to maximize classification accuracy. Then, the current population undergoes genetic operations (i.e. selection, mutation and crossover) and a new population is generated and evaluated. This evolution process is repeated until a pre-defined number of generations is reached. It outputs the "best individual" i.e. the best predictor for the considered subspace. Since the GA performs a stochastic search, it is applied T times on each of the R subspaces.

The fourth and final task is to extract domain knowledge by the T·R predictors obtained at the previous layer. This is done by analyzing the frequency of membership of each feature in the collected predictors. Such analysis enables evaluating the relative importance of each feature, distinguishing the features that play a primary role in discriminating the target class from those that give a complementary, yet not negligible, contribution.

## MATERIALS AND METHODS

The framework evaluation can be supported by a variety of popular ranking techniques and classification algorithms. In what follows, we detail the specific methods and settings that we adopted to implement the different layers of the framework. Moreover, we give a brief description of the datasets used in the experimental validation.

### Ranking and Filtering

Being supported by our previous experience (Cannas, Dessì, & Pes, 2011; Dessì, Milia, & Pes, 2013), we choose *Chi Squared* ($\chi^2$) as ranking metric (layer 1). Basically, $\chi^2$ evaluates features individually by measuring their chi-squared statistic with respect to the class: the larger the chi-squared, the more important a feature is for the classification task at hand (Liu & Setiono, 1995). In more detail, once a feature has been discretized into a number $I$ of intervals, its $\chi^2$ value is computed as:

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{C} \frac{\left(A_{ij} - \frac{R_i \cdot B_j}{M}\right)^2}{\frac{R_i \cdot B_j}{M}}$$

where $M$ is the number of instances, $C$ the number of classes, $R_i$ the number of instances in the $i$th interval, $B_j$ the number of instances in the $j$th class, and $A_{ij}$ the number of instances in the $i$th interval and $j$th class. The effectiveness of this approach for ranking features has been proved in different domains (Forman, 2003; Yang and Pedersen, 1997).

The resulting ranked list provides the basis for the incremental filtering step (layer 2). Here, the framework parameters are set as P = 10 and R = 5. Specifically, starting from the subset including the first P ranked features, namely the subset TOP10, we constructed R-1 additional nested subsets of features of increasing size by

progressively adding P features (less and less correlated with the target). We denote these additional subspaces as TOP20 (i.e. the first 20 top-ranked features), TOP30 (i.e. the first 30 top-ranked features), etc. We also considered TOP80 and TOP100 in order to evaluate the proposed approach in larger feature subspaces.

## Genetic Search

At the third layer of the framework, the wrapper is based on the GA search mechanism as proposed by Goldberg (1989). As previously mentioned, at the start of the GA, a population of individuals (i.e. feature subsets) is initialized randomly. In more detail, these individuals are encoded by N-bit binary vectors (where N is the size of the search space). If a bit is '1' it means that the corresponding feature is included in the subset, while the bits with value '0' mean the opposite.

The merit a feature subset $x$ is evaluated by a fitness function $F(x)$ that expresses the classification accuracy of a classifier built on $x$. In our experiments, we considered two popular classifiers, i.e. Support Vector Machines (SVM), with linear kernel, and K-Nearest Neighbor (K-NN), with K = 1. This resulted in two different implementations of the genetic wrapper, namely GA/SVM and GA/K-NN. Error estimation was performed by a 10-fold cross-validation for both SVM and K-NN classifiers.

As regards the genetic operators, we relied on the following well established settings:

- **Selection:** *Roulette wheel selection* is used to probabilistically select individuals from a population for later breeding. The probability $P(x_i)$ of selecting the individual $x_i$ is proportional to its own fitness $F(x_i)$ and inversely proportional to the fitness of other competing hypotheses in the current population:

$$P(x_i) = \frac{F(x_i)}{\sum_i F(x_i)}$$

- **Crossover:** We use the *single point crossover*, i.e. one crossover point $i$ is chosen at random so that the first $i$ bits are contributed by one parent and the remaining bits by the second parent;
- **Mutation:** Each individual has a probability $p_m$ to mutate. We randomly choose a number of $n$ bits to be flipped in every mutation stage.

As regards the stopping criterion, the genetic search ends when a pre-defined number of generations is reached or a fitness value of 100% is obtained.

Leveraging on previous studies about tuning GA parameters (Cannas, Dessì, & Pes, 2010), we set the following values: population size = 30, number of generations = 50, probability of crossover = 1, and probability of mutation = 0.02. Moreover, since the GA performs a stochastic search, we considered the average results over a number T = 10 of trials; indeed, this is a common choice in literature (Jirapech-Umpai & Aitken, 2005).

## Datasets and Related Experiments

We chose to experiment with high dimensional data from genomics datasets of DNA-microarray experiments. A short description of these datasets is given in Table 1.

We worked with four different datasets: *Leukemia* (Golub et al., 1999), where the goal is to distinguish between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL); *DLBCL* (Shipp et al., 2002), where the goal is to recognize diffuse large b-cell lymphoma (DLBCL) as different from follicular lymphoma (FL); *Colon* (Alon et al., 1999), where the goal is to distinguish between healthy and tumor colon tissues; *Prostate* (Singh et al., 2002), where the goal is to distinguish between healthy and tumor prostate tissues. Features correspond to levels of expression of different genes and are continuous.

We evaluated the proposed framework on each dataset by performing two experiments.

*Table 1. Microarray datasets used in the experiments*

| Dataset | No. of Samples | Distribution among Classes | No. of Features | Reference |
|---|---|---|---|---|
| Leukemia | 72 | 47 ALL + 25 AML | 7129 | Golub et al., 1999 |
| DLBCL | 78 | 58 DLBCL + 19 FL | 7129 | Shipp et al., 2002 |
| Colon | 62 | 40 tumor + 22 normal | 2000 | Alon et al., 1999 |
| Prostate | 102 | 52 tumor + 50 normal | 10509 | Singh et al., 2002 |

In the first experiment, namely the *baseline experiment*, each classifier (i.e. K-NN and SVM) was trained directly on each subspace TOPN in order to estimate the accuracy without a wrapper based feature selection. This *baseline accuracy* was also estimated by a 10-fold cross-validation. The second experiment evaluated the effectiveness of the wrappers GA/SVM and GA/K-NN on each subspace TOPN. The overall experimentation leverages on the WEKA machine learning environment (Bouckaert et al., 2010).

## RESULTS AND DISCUSSION

In this section, the results of our experiments will be discussed along two dimensions: (1) the effectiveness of the proposed framework in searching suitable combinations of relatively few features that yield high classification accuracy; (2) the representation level reached by the framework in exploring how each feature may be useful in representing essential knowledge about the application domain.

### Effectiveness of the Framework

We compare first the differences between the baseline accuracy and the accuracy (best and average) reached by GA/K-NN and GA/SVM on each TOPN. Tables 2 through 5 report this comparison for each dataset.

As Table 2 (Leukemia) shows, results produced by both GA/SVM and GA/K-NN outperform baseline results from SVM and K-NN. The average accuracy of GA/SVM increases with the size of the search space until reaching 100% on TOP80 and TOP100. GA/K-NN turns out to be more effective in selecting feature subsets that perfectly discriminate the target class (namely, perfect predictors), irrespective of the size of TOPN: a search space of

*Table 2. Baseline, average and best accuracy (leukemia)*

| | SVM | GA/SVM | | K-NN | GA/K-NN | |
|---|---|---|---|---|---|---|
| | Baseline Accuracy (%) | Average Accuracy (%) | Best Accuracy (%) | Baseline Accuracy (%) | Average Accuracy (%) | Best Accuracy (%) |
| TOP10 | 93.1 | 97.0 | 97.5 | 91.7 | 100 | 100 |
| TOP20 | 95.8 | 99.3 | 100 | 97.2 | 100 | 100 |
| TOP30 | 98.6 | 99.6 | 100 | 94.4 | 100 | 100 |
| TOP40 | 98.6 | 99.9 | 100 | 95.8 | 100 | 100 |
| TOP50 | 97.2 | 99.4 | 100 | 93.1 | 99.9 | 100 |
| TOP80 | 97.2 | 100 | 100 | 95.8 | 100 | 100 |
| TOP100 | 97.2 | 100 | 100 | 97.2 | 100 | 100 |

*Table 3. Baseline, average and best accuracy (b) DLBCL*

| | SVM | GA/SVM | | K-NN | GA/K-NN | |
|---|---|---|---|---|---|---|
| | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** |
| TOP10 | 92.2 | 92.5 | 92.7 | 85.7 | 93.8 | 94.3 |
| TOP20 | 94.8 | 96.8 | 97.4 | 93.5 | 99.9 | 100 |
| TOP30 | 94.8 | 98.1 | 98.7 | 96.1 | 100 | 100 |
| TOP40 | 96.1 | 98.7 | 100 | 94.8 | 99.7 | 100 |
| TOP50 | 96.1 | 98.1 | 98.7 | 94.8 | 100 | 100 |
| TOP80 | 97.4 | 99.1 | 100 | 96.1 | 100 | 100 |
| TOP100 | 94.8 | 100 | 100 | 96.1 | 99.9 | 100 |

*Table 4. Baseline, average and best accuracy (c) Colon*

| | SVM | GA/SVM | | K-NN | GA/K-NN | |
|---|---|---|---|---|---|---|
| | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** |
| TOP10 | 82.3 | 87.1 | 87.1 | 80.6 | 90.8 | 91.3 |
| TOP20 | 88.7 | 90.9 | 91.9 | 82.3 | 95.3 | 98.4 |
| TOP30 | 87.1 | 90.5 | 91.9 | 83.9 | 94.5 | 95.2 |
| TOP40 | 85.5 | 91.5 | 92.3 | 83.9 | 94.5 | 96.8 |
| TOP50 | 83.9 | 91.5 | 91.9 | 80.6 | 92.7 | 95.2 |
| TOP80 | 85.5 | 91.9 | 93.2 | 79.0 | 94.6 | 96.8 |
| TOP100 | 87.1 | 93.1 | 94.2 | 79.0 | 93.2 | 95.5 |

*Table 5. Baseline, average and best accuracy (d) Prostate*

| | SVM | GA/SVM | | K-NN | GA/K-NN | |
|---|---|---|---|---|---|---|
| | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** | **Baseline Accuracy (%)** | **Average Accuracy (%)** | **Best Accuracy (%)** |
| TOP10 | 95.1 | 95.4 | 95.7 | 92.2 | 94.4 | 94.7 |
| TOP20 | 96.1 | 96.6 | 97.1 | 93.1 | 96.2 | 97.1 |
| TOP30 | 94.1 | 97.8 | 98.0 | 89.2 | 97.9 | 98.0 |
| TOP40 | 97.1 | 97.8 | 98.0 | 93.1 | 98.0 | 98.0 |
| TOP50 | 96.1 | 97.9 | 98.0 | 94.1 | 98.0 | 98.0 |
| TOP80 | 96.1 | 97.3 | 98.0 | 92.2 | 98.0 | 98.0 |
| TOP100 | 96.1 | 97.2 | 98.0 | 90.2 | 98.0 | 98.0 |

10 features is sufficient to reach the maximum accuracy that is also reached in all the feature subspaces with the exception of TOP50.

Table 3 shows the same trend for experiments on DLBCL dataset, both with GA/SVM and GA/K-NN.

According to Table 4, neither GA/SVM nor GA/K-NN are able to find perfect predictors in the Colon dataset. However, the average accuracy of GA/SVM exhibits the same behavior than in previous experiments. The effectiveness of the GA/K-NN is confirmed, regardless of the size of the feature subspace: the best predictor is extracted from TOP20.

Finally, Table 5 reports results about Prostate dataset and shows a picture quite different from the three previous datasets. The trend of the average accuracy of GA/SVM reaches the highest value between TOP30 and TOP50 and then starts decreasing. GA/K-NN outperforms GA/SVM very slightly, since the values of the average accuracy are highly similar and, in addition, both achieve the same values of best accuracy.

Globally, results in Tables 2 through 5 confirm that the classification can be carried out in a reduced space more accurately that in the original feature subspace as the use of an unnecessarily large gene set may decrease the effectiveness in the classification process.

By showing the trend of the average size of selected predictors as the size of the feature subspace TOPN increases, Figure 2 demonstrates the effectiveness of the framework in reducing the dimensionality of the search space. In particular, both GA/SVM and GA/K-NN considerably cut the size of the original TOPN whose average reduction is greater than 50% with peaks of 70-75% reached on TOP100. This trend is common to all the datasets.

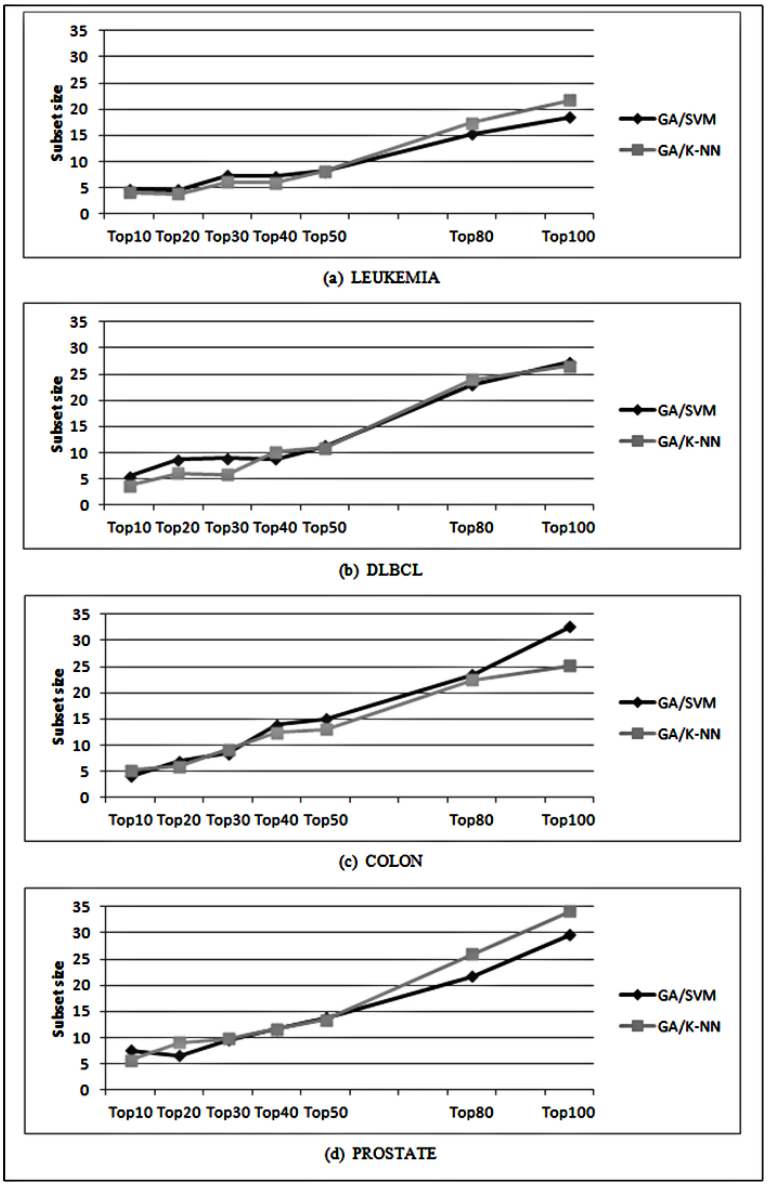
As Table 6 shows, this reduction generates sufficient features for achieving a very high accuracy. In detail, for the best predictors selected by GA/SVM and GA/K-NN on each dataset, Table 6 summarizes the accuracy, the minimum size and the feature space from which the predictor was extracted. For "good" datasets such as Leukemia and DLBCL, the framework generates perfect predictors. For more difficult datasets, such as Colon and Prostate, the framework does not achieve the 100% accuracy albeit obtaining remarkable results.

Globally, the results shown in Table 6 help to demonstrate the framework effectiveness and can be compared with those produced by different methods in DNA-microarray literature. As reference parameters, we considered the accuracy and the number of selected features. We present the best results achieved by GA/K-NN and omit results from GA/SVM that, except for the Colon dataset, exhibit the same trend. Tables 7 through 10 show this comparison.

Regarding Leukemia dataset (Table 7), different methods proposed in literature achieve 100% of accuracy, as in our approach, but the number of features they select is greater than the one obtained by GA/K-NN. Our method shows excellent performance also in DLBCL dataset, as shown in Table 8. Regarding Colon (Table 9) and Prostate (Table 10) datasets, which are recognized as more challenging benchmarks, our results are in turn superior or comparable to those reported in literature.

We conclude this discussion on the effectiveness of the framework with some considerations about the computational cost. Indeed, one of the well-known drawbacks of genetic approaches is that they usually require high execution times when applied to high dimensional search spaces. To mitigate this problem, our framework reduces the dimensionality of the original dataset by defining proper subspaces of pre-filtered features. Indeed, this enables us to reach a good trade-off between effectiveness and computational cost. For example, in the case of Leukemia dataset, the average execution times of GA/SVM (over 10 trials) range from 55 seconds for TOP10 to 7 minutes for TOP100, while the average execution times of GA/K-NN range from 2 seconds for TOP10 to 26 seconds for TOP100. Of course, baseline execution times (without a genetic search) are

*Figure 2. Average size of the selected predictors as the size of the feature space increases*




remarkable lower (less than 0.1 seconds, for both SVM and K-NN)[1]. For the other datasets here considered, we observed a very similar trend, i.e. for both GA/SVM and GA/K-NN the computational cost increases with the size of the search space but the execution times of GA/K-NN are considerably lower. Hence, the framework efficiency may sensibly vary depending on the chosen classification algorithm. In particular, GA/KNN seems to be a very good option since it leads to a very effective feature selection in a quite efficient way.

*Table 6. Best predictors extracted by GA/SVM and GA/K-NN from each dataset*

| | Wrapper | Search Space | Subset Size | Accuracy (%) |
|---|---|---|---|---|
| Leukemia | GA/SVM | TOP20 | 4 | 100 |
| | GA/K-NN | TOP20 | 3 | 100 |
| | | TOP30 | 3 | 100 |
| | | TOP40 | 3 | 100 |
| DLBCL | GA/SVM | TOP40 | 8 | 100 |
| | GA/K-NN | TOP20 | 4 | 100 |
| | | TOP30 | 4 | 100 |
| Colon | GA/SVM | TOP100 | 39 | 94.2 |
| | GA/K-NN | TOP20 | 4 | 98.4 |
| Prostate | GA/SVM | TOP30 | 8 | 98.0 |
| | GA/K-NN | TOP30 | 8 | 98.0 |
| | | TOP40 | 8 | 98.0 |

*Table 7. Framework performance vs different methods in literature (leukemia)*

| | GA /K-NN | Huerta et al., 2006 | Wang et al., 2005 | Leung & Hung, 2010 | Ng & Chan, 2005 |
|---|---|---|---|---|---|
| **Accuracy (%)** | 100 | 100 | 100 | 100 | 100 |
| **Subset size** | 3 | 25 | 8 | 4 | 4 |

*Table 8. Framework performance vs different methods in literature (DLBCL)*

| | GA /K-NN | Liu & Zhou, 2003 | Leung & Hung, 2010 | Dagliyan et al., 2011 | Deutsch, 2003 |
|---|---|---|---|---|---|
| **Accuracy (%)** | 100 | 93.5 | 100 | 96.1 | 100 |
| **Subset size** | 4 | 5 | 6 | 6 | 4 |

*Table 9. Framework performance vs different methods in literature (colon)*

| | GA /K-NN | Reddy & Deb, 2003 | Huerta et al., 2006 | Leung & Hung, 2010 | Yu & Liu, 2004 |
|---|---|---|---|---|---|
| **Accuracy (%)** | 98.4 | 97.0 | 99.4 | 95.2 | 93.6 |
| **Subset size** | 4 | 7 | 10 | 6 | 4 |

*Table 10. Framework performance vs different methods in literature (prostate)*

| | GA /K-NN | Zhang & Deng, 2007 | Dagliyan et al., 2011 | Leung & Hung, 2010 | Küçükural et. al, 2007 |
|---|---|---|---|---|---|
| **Accuracy (%)** | 98.0 | 96.1 | 96.1 | 98.0 | 96.7 |
| **Subset size** | 8 | 13 | 11 | 6 | 19 |

## Representation Level of the Framework

Domain experts instinctively have high confidence in the results of a selection method that finds similar sets of features: the fact that a gene is selected by different predictors makes it more probable that this gene is an important biomarker. Hence, we assume the frequency of each gene in the selected predictors as a measure of the representation level of the framework.

Specifically, for each microarray dataset we evaluated the frequency of the features belonging to the 70 predictors (i.e. 10 predictors for each of the 7 considered subspaces) obtained at layer 3. For each dataset, Tables 11 through 14 show the frequency of the ten most selected features and reports, in brackets, the position of each feature in the original ranked list obtained at layer 1.

Analyzing the features that are most frequently selected by GA/SVM and GA/K-NN, we note, in Table 11 (Leukemia), that the two lists have 7 features in common out of 10. Besides, features that appear only in GA/SVM list are also selected by GA/K-NN with lower frequency, and vice versa. Further, we notice that the features most frequently involved in the selected predictors are not necessarily the top-ranked ones: for example, the gene 1928 exhibits the highest frequency for GA/SVM but it is placed at ranking position 30; likewise, the most frequent gene for GA/K-NN, the gene 2354, is placed at ranking position 14. As well, genes 4951 and 5107 exhibit a very high frequency, but they are at ranking position 67 and 68, respectively. In turn, some top-ranked genes such as 3252 and 2288 do not appear at all in the two lists even if they are respectively at positions 4 and 6 of the ranked list.

*Table 11. Frequency of the ten most selected features; in brackets, the position of each feature in the original ranked list (leukemia)*

| GA/SVM | | GA/K-NN | |
|---|---|---|---|
| **Features** | **Frequency** | **Features** | **Frequency** |
| 1928 (30) | 47.1% | 2354 (14) | 44.3% |
| 1144 (17) | 41.4% | 1834 (1) | 42.9% |
| 2354 (14) | 38.6% | 6855 (5) | 38.6% |
| 6855 (5) | 37.1% | 1928 (30) | 38.6% |
| 1685 (9) | 37.1% | 1685 (9) | 37.1% |
| 1834 (1) | 35.7% | 804 (31) | 32.9% |
| 4847 (2) | 34.3% | 1144 (17) | 31.4% |
| 804 (31) | 30.0% | 1882 (3) | 24.3% |
| 2020 (22) | 24.3% | 5107 (68) | 24.3% |
| 2642 (28) | 22.9% | 4951 (67) | 20.0% |

Table 12 reports results regarding DLBCL dataset. In this case the two lists have 8 features in common. Genes 1670 and 5077 exhibit the highest frequencies for both GA/SVM and GA/K-NN even if they are placed at ranking positions 18 and 29. On the contrary, genes at ranking positions 2 and 3 do not appear in the lists.

As Table 13 (Colon) reports, the two lists have 5 features in common. Features that exhibit the highest frequency are gene 66 for GA/SVM and gene 1772 for GA/K-NN and are placed at ranking position 15 and 10, respectively. Again, some top-ranked genes, such as those at positions 1 and 2, do not emerge.

Table 14 reports results regarding Prostate dataset. The two lists have 7 features in common. Features that exhibit the highest frequency, for both GA/SVM and GA/K-NN, are 4823, 10130, and 9138 and are placed at ranking positions 1, 13, and 16. Although features at ranking positions 1 and 2 are present, genes at ranking position 3 and 4 do not appear in the lists.

Therefore, in each of the considered case studies, the high number of features in common between the two lists shows that GA/SVM and GA/K-NN highly agree in evaluating the relevance of features, although selecting different gene combinations (at layer 3). This suggests that the proposed framework can be useful to evaluate the relative importance of features in a context where multiple predictors may coexist, such as microarray data classification.

Another remarkable aspect to note is that the feature lists obtained at layer 4 do not match the ranked list produced as output of the first layer; indeed the features most frequently selected are not necessarily the top-ranked ones. This outlines that, while useful in reducing the dimensionality of the initial problem, the ranking process is not by itself a suitable feature selection technique for microarray data. It can be successfully employed, instead, within hybrid filter-wrapper approaches as the one proposed here.

## RELATED WORK

The problem of feature selection has received a thorough treatment in machine learning and pattern recognition. Many surveys attempt to review the field (Bolón-Canedo, Sánchez-Maroño, & Alonso-Betanzos, 2013; Chandrashekar & Sahin, 2014; Kumar & Minz, 2014; Tang et al., 2014). In the specific context of bioinformatics applications, a comprehensive review of feature selection techniques is provided by Saeys and

*Table 12. Frequency of the ten most selected features; in brackets, the position of each feature in the original ranked list (DLBCL)*

| GA/SVM | | GA/K-NN | |
|---|---|---|---|
| **Features** | **Frequency** | **Features** | **Frequency** |
| 5077 (29) | 70.0% | 1670 (18) | 65.7% |
| 1670 (18) | 64.3% | 5077 (29) | 37.1% |
| 4453 (11) | 45.7% | 3818 (32) | 37.1% |
| 3005 (24) | 42.9% | 4453 (11) | 34.3% |
| 506 (1) | 41.4% | 373 (12) | 34.3% |
| 203 (13) | 41.4% | 1055 (9) | 32.9% |
| 373 (12) | 37.1% | 2789 (38) | 31.4% |
| 2789 (38) | 37.1% | 506 (1) | 30.0% |
| 3818 (32) | 35.7% | 3005 (24) | 30.0% |
| 4202 (5) | 32.9% | 6493 (43) | 30.0% |

*Table 13. Frequency of the ten most selected features; in brackets, the position of each feature in the original ranked list (colon)*

| GA/SVM | | GA/K-NN | |
|---|---|---|---|
| **Features** | **Frequency** | **Features** | **Frequency** |
| 66 (15) | 64.3% | 1772 (10) | 75.7% |
| 493 (3) | 61.4% | 765 (4) | 72.9% |
| 1423 (5) | 60.0% | 1423 (5) | 41.4% |
| 1771 (6) | 58.6% | 267 (8) | 35.7% |
| 1772 (10) | 57.1% | 415 (21) | 35.7% |
| 897 (14) | 52.9% | 513 (7) | 34.3% |
| 1042 (19) | 48.6% | 1892 (20) | 34.3% |
| 765 (4) | 47.1% | 1771 (6) | 32.9% |
| 581 (35) | 45.7% | 897 (14) | 32.9% |
| 780 (12) | 42.9% | 822 (16) | 32.9% |

*Table 14. Frequency of the ten most selected features; in brackets, the position of each feature in the original ranked list (prostate)*

| GA/SVM | | GA/K-NN | |
|---|---|---|---|
| **Features** | **Frequency** | **Features** | **Frequency** |
| 4823 (1) | 67.1% | 4823 (1) | 67.1% |
| 10130 (13) | 67.1% | 9138 (16) | 67.1% |
| 2718 (30) | 60.0% | 7346 (8) | 65.7% |
| 7652 (5) | 57.1% | 7652 (5) | 61.4% |
| 9138 (16) | 57.1% | 3997 (27) | 54.3% |
| 7515 (21) | 51.4% | 3124 (39) | 54.3% |
| 8765 (2) | 48.6% | 8765 (2) | 51.4% |
| 8009 (7) | 47.1% | 8009 (7) | 51.4% |
| 1943 (23) | 47.1% | 10130 (13) | 45.7% |
| 5648 (25) | 42.9% | 2718 (30) | 45.7% |

al. (2007). As well, representative algorithms for feature selection are empirically evaluated, under different problem settings and from different perspectives, by recent comparative studies (Abusamra, 2013; Dessì, Pascariello, & Pes, 2013; Khoshgoftaar, Gao, Napolitano, & Wald, 2014; Staroszczyk, Osowski, & Markiewicz, 2012).

Since 2001, a significant effort has been done to develop new and adapt known feature selection techniques in the context of microarray datasets (Bolón-Canedo et al., 2014). In particular, because of the high dimensionality of most microarray analyses, the filter model is often preferred in gene selection due to its computational efficiency. Characteristics and

particularities of filter techniques for microarray data are extensively discussed by Lazar and al. (2012). With regard to the wrapper model, most methods use randomized search heuristics, although also a few examples adopt sequential search techniques (Saeys et al., 2007). Hybrid and more sophisticated feature selection techniques have been explored in recent microarray research efforts (Leung, & Hung, 2010). In hybrid models, the key is to initially reduce the search space using a filter method and subsequently apply wrapper methods, hence reducing the computation time.

Similarly to our approach, evolutionary algorithms have been applied to microarray analysis in order to look for the optimal or near optimal sets of predictive genes (Jirapech-Umpai & Aitken, 2005). For example, Huerta and al. (2006) address the problem of gene selection using a standard genetic algorithm which evolves populations of possible solutions, the quality of each solution being evaluated by an SVM classifier. Genetic algorithms have been employed in conjunction with different classifiers, such as K-Nearest Neighbor (Lee, Lin, Chen, & Kuo, 2011) and Neural Networks (Bevilacqua, Mastronardi, Menolascina, Paradiso, & Tommasi, 2006).

## CONCLUSION

The key idea of this paper is to balance two basic aspects of feature selection, i.e. effectiveness and representation level, in order to achieve a two-fold objective: finding good predictors for effective classification and providing a representation of the application domain with a model that fits as much as possible the reality.

According to such idea, the paper has presented a hybrid framework which combines rankers, filters and wrappers methods in a multi-layer and modular architecture. Each layer involves a general class of methods by privileging, in turn, one of the two above aspects. This allows the framework to be loosely coupled with the specific algorithms chosen for its implementation.

The proposed framework has been validated on several microarray DNA-datasets and experimental results compare well with different hybrid methods proposed in literature. Results show that our approach is effective in finding small subsets of informative features with high classification accuracy and suitable representation level. In addition, results suggest that the framework is able to significantly evaluate the relative importance of features in those contexts where multiple predictors may coexist, such as DNA-microarray data classification.

As future work, we will verify the proposed framework by considering a variety of high-dimensional datasets from different application domains. In particular, preliminary experiments on text categorization seem to suggest that the framework is suitable for text-based data.

## ACKNOWLEDGMENT

This research was supported by RAS, Regione Autonoma della Sardegna (Legge regionale 7 agosto 2007, n. 7 "Promozione della ricerca scientifica e dell'innovazione tecnologica in Sardegna") in the project "*DENIS: Dataspaces Enhancing the Next Internet in Sardinia*".

## REFERENCES


Abusamra, H. (2013). A Comparative Study of Feature Selection and Classification Methods for Gene Expression Data of Glioma. *Procedia Computer Science*, *23*, 5–14. doi:10.1016/j.procs.2013.10.003

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. (1999). Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays. *Proceedings of the National Academy of Sciences of the United States of America*, *96*(12), 6745–6750. doi:10.1073/pnas.96.12.6745 PMID:10359783

Bevilacqua, V., Mastronardi, G., Menolascina, F., Paradiso, A., & Tommasi, S. (2006). Genetic algorithms and artificial neural networks in microarray data analysis: A distributed approach. *Engineering Letters*, *13*(3), 335–343.

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2013). A review of feature selection methods on synthetic data. *Knowledge and Information Systems*, *34*(3), 483–519. doi:10.1007/s10115-012-0487-8

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., Benítez, J. M., & Herrera, F. (2014). A review of microarray datasets and applied feature selection methods. *Information Sciences*, *282*, 111–135. doi:10.1016/j.ins.2014.05.042

Bouckaert, R. R., Frank, E., Hall, M. A., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2010). WEKA - Experiences with a Java Open-Source Project. *Journal of Machine Learning Research*, *11*, 2533–2541.

Cannas, L. M., Dessì, N., & Pes, B. (2010). A filter-based evolutionary approach for selecting features in high-dimensional micro-array data. In IFIP Advances in Information and Communication Technology, IFIP AICT 340 (pp. 297–307). doi:10.1007/978-3-642-16327-2_36

Cannas, L. M., Dessì, N., & Pes, B. (2011). Knowledge Discovery in Gene Expression Data via Evolutionary Algorithms. In *Proceedings of the 2nd International Workshop on Biological Knowledge Discovery and Data Mining, BIOKDD '11, DEXA 2011 Workshops* (pp. 402-406). doi:10.1109/DEXA.2011.48

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, *40*(1), 16–28. doi:10.1016/j.compeleceng.2013.11.024

Dagliyan, O., Uney-Yuksektepe, F., Kavakli, I. H., & Turkay, M. (2011). Optimization Based Tumor Classification from Microarray Gene Expression Data. *PLoS ONE*, *6*(2), e14579. doi:10.1371/journal.pone.0014579 PMID:21326602

Dessì, N., Milia, G., & Pes, B. (2013). Enhancing Random Forests Performance in Microarray Data Classification. In *Proceedings of the 14th Conference on Artificial Intelligence in Medicine, AIME 2013 (LNAI)* (Vol. 7885, pp. 99-103). doi:10.1007/978-3-642-38326-7_15

Dessì, N., Pascariello, E., & Pes, B. (2013). A Comparative Analysis of Biomarker Selection Techniques. BioMed Research International, 2013, Article ID 387673.

Deutsch, J. M. (2003). Evolutionary algorithms for finding optimal gene sets in microarray prediction. *Bioinformatics (Oxford, England)*, *19*(1), 45–52. doi:10.1093/bioinformatics/19.1.45 PMID:12499292

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, *3*, 1289–1305.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., & Lander, E. S. et al. (1999). Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*, *286*(5439), 531–537. doi:10.1126/science.286.5439.531 PMID:10521349

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Huerta, E. B., Duval, B., & Hao, J. K. (2006). A hybrid GA/SVM approach for gene selection and classification of microarray data. *In Proceedings of EvoWorkshops 2006 (LNCS)* (Vol. 3907, pp. 34–44). doi:10.1007/11732242_4

Jirapech-Umpai, T., & Aitken, S. (2005). Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics*, *6*(1), 148. doi:10.1186/1471-2105-6-148 PMID:15958165

Khoshgoftaar, T. M., Gao, K., Napolitano, A., & Wald, R. (2014). A comparative study of iterative and non-iterative feature selection techniques for software defect prediction. *Information Systems Frontiers*, *16*(5), 801–822. doi:10.1007/s10796-013-9430-0

Küçükural, A., Yeniterzi, R., Yeniterzi, S., & Sezerman, O. U. (2007). Evolutionary selection of minimum number of features for classification of gene expression data using genetic algorithms. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation* (pp. 401-406). doi:10.1145/1276958.1277040

Kumar, V., & Minz, S. (2014). Feature Selection: A literature Review. *Smart Computing Review*, *4*(3), 211–229. doi:10.1145/2740070.2626320

Lazar, C., Taminau, J., Meganck, S., Steenhoff, D., Coletta, A., Molter, C., & Nowé, A. et al. (2012). A Survey on Filter Techniques for Feature Selection in Gene Expression Microarray Analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *9*(4), 1106–1119. doi:10.1109/TCBB.2012.33 PMID:22350210

Lee, C. P., Lin, W. S., Chen, Y. M., & Kuo, B. J. (2011). Gene selection and sample classification on microarray data based on adaptive genetic algorithm/k-nearest neighbor method. *Expert Systems with Applications*, *38*(5), 4661–4667. doi:10.1016/j.eswa.2010.07.053

Leung, Y., & Hung, Y. (2010). A multiple-filter-multiple-wrapper approach to gene selection and microarray data classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *7*(1), 108–117. doi:10.1109/TCBB.2008.46 PMID:20150673

Liu, H., & Setiono, R. (1995). Chi2: Feature selection and discretization of numeric attributes. In *Proceedings of the 7th International Conference on Tools with Artificial Intelligence, ICTAI'95* (pp. 338-391).

Liu, H., & Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, *17*(3), 1–12.

Liu, J., & Zhou, H. B. (2003). Tumor Classification Based on Gene Microarray Data and Hybrid Learning Method. In *Proc. Int'l Conf. Machine Learning and Cybernetics* (pp. 2275-2280).

Ng, M., & Chan, L. (2005). Informative Gene Discovery for Cancer Classification from Microarray Expression Data. In *Proc. IEEE Workshop Machine Learning for Signal Processing, MLSP '05* (pp. 393-398).

Reddy, A. R., & Deb, K. (2003). Classification of two-class cancer data reliably using evolutionary algorithms. *Bio Systems*, *72*, 111–129. doi:10.1016/S0303-2647(03)00138-2 PMID:14642662

Saeys, Y., Inza, I., & Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics (Oxford, England)*, *23*(19), 2507–2517. doi:10.1093/bioinformatics/btm344 PMID:17720704

Shipp, M. A., Ross, K. N., Tamayo, P., Weng, A. P., Kutok, J. L., Aguiar, R. C., & Golub, T. R. et al. (2002). Diffuse Large B-cell Lymphoma Outcome Prediction by Gene-expression Profiling and Supervised Machine Learning. *Nature Medicine*, *8*(1), 68–74. doi:10.1038/nm0102-68 PMID:11786909

Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., & Sellers, W. R. et al. (2002). Gene Expression Correlates of Clinical Prostate Cancer Behavior. *Cancer Cell*, *1*(2), 203–209. doi:10.1016/S1535-6108(02)00030-2 PMID:12086878

Staroszczyk, T., Osowski, S., & Markiewicz, T. (2012). Comparative Analysis of Feature Selection Methods for Blood Cell Recognition in Leukemia. In *Proceedings of the 8th International Conference on Machine Learning and Data Mining in Pattern Recognition, MLDM 2012* (pp 467-481). doi:10.1007/978-3-642-31537-4_37

Tang, J., Alelyani, S., & Liu, H. (2014). Feature Selection for Classification: A Review. In C. C. Aggarwal (Ed.), *Data Classification: Algorithms and Applications* (pp. 37–64). CRC Press.

Wang, Y., Makedon, F., Ford, J. C., & Pearlman, J. D. (2005). Hykgene: A hybrid approach for selecting marker genes for phenotype classification using microarray gene expression data. *Bioinformatics (Oxford, England)*, *21*(8), 1530–1537. doi:10.1093/bioinformatics/bti192 PMID:15585531

Yang, Y., & Pedersen, J. O. (1997). A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning, ICML '97* (pp. 412-420).

Yu, L., & Liu, H. (2004). Redundancy Based Feature Selection for Microarray Data. In *Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, KDD '04* (pp. 737-742). doi:10.1145/1014052.1014149

Zhang, J. G., & Deng, H. W. (2007). Gene selection for classification of microarray data based on the bayes error. *BMC Bioinformatics*, *8*(1), 370. doi:10.1186/1471-2105-8-370 PMID:17915022

## ENDNOTES

[1] Experiments were performed with a CPU of 2.8 GHz and 4GB of RAM.

*Nicoletta Dessi is associated professor of Computer Science at University of Cagliari, where she currently teaches two Database courses (Fundamental and Advanced). She received the Ph.D. degree in Mathematics from University of Cagliari, in 1970. From 2004-2010 she has been deputy head of the Department of Mathematics and Computer Science at University of Cagliari. Her main research interests are in the area of data mining and information systems, with current focus on bioinformatics. She is also involved in data integration research based on Cloud, Web Services, and trusted cooperation. She has published her research results in international journals and the refereed proceedings of the major conferences of the field. Currently, she is the principal investigator and the responsible of the research project "DENIS: Dataspaces Enhancing the Next Internet in Sardinia", funded by RAS, Regione Autonoma della Sardegna (Italy) in 2012.*

*Barbara Pes was born in Cagliari, Italy, in 1976. She obtained her laurea degree in Physics from the University of Cagliari in 2001. From 2002 to 2005 she collaborated with the Database and Data Mining Group at the Department of Mathematics and Computer Science, University of Cagliari. Since 2006 she has been working as Assistant Professor (Ricercatore Universitario) at the Science Faculty, University of Cagliari, where she teaches Foundations of Computer Science and Data Mining courses. She is author of more than 30 papers published in international conferences, books and journals. Her research interests include Service Oriented Architectures, Data Integration, Data Mining and Knowledge Discovery in Databases, Feature Selection and Classification of High-dimensional Data, Bio-informatics.*

*Laura Maria Cannas was born in Iglesias, Italy, in 1984. She obtained her Ph.D. in Computer Science from the University of Cagliari in 2013. Her research activity focuses on data mining and knowledge discovery in high-dimensional datasets. Specifically, her interests include feature selection techniques and classification of biomedical data.*